Video-Based Motion Retargeting Framework between Characters with Various Skeleton Structure

Xin Huang
The University of Tokyo
Tokyo, Japan
xhuang1019@gmail.com

Takashi Kanai
The University of Tokyo
Tokyo, Japan
kanait@acm.org

ABSTRACT

We introduce a motion retargeting framework capable of animating characters with distinct skeletal structures using video data. While prior studies have successfully performed motion retargeting between skeletons with different structures, retargeting noisy and unnatural motion data extracted from monocular videos has proved challenging. Addressing this issue, our approach proposes a deep learning framework, retargeting motion data procured from easily accessible monocular videos, to animate characters with diverse skeletal structures. Our approach is aimed at providing support for individual creators in character animation.

Our proposed framework pre-processes motion data derived from multiple monocular videos by two-stage pose estimation, using this as the training dataset for Skeleton-Aware Motion Retargeting Network (SAMRN). In addition, we introduce a loss function for the rotation angle of the character's root node to address the rotation issue inherent in SAMRN. Furthermore, by incorporating motion data extracted from videos and adding a loss function for the character's root node and end-effector's velocities, the proposed method makes it possible to generate natural motion data that is closely aligned with the source video. We demonstrate the effectiveness of the proposed framework for motion retargeting between monocular videos and various characters through both qualitative and quantitative evaluations.

CCS CONCEPTS

• **Computing methodologies** → **Motion processing**.

KEYWORDS

motion retargeting, neural networks, character animation

ACM Reference Format:
Xin Huang and Takashi Kanai. 2023. Video-Based Motion Retargeting Framework between Characters with Various Skeleton Structure. In *ACM SIGGRAPH Conference on Motion, Interaction and Games (MIG '23), November 15–17, 2023, Rennes, France.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3623264.3624473

MIG '23, November 15–17, 2023, Rennes, France

ACM ISBN 979-8-4007-0393-5/23/11…$15.00
https://doi.org/10.1145/3623264.3624473

1 INTRODUCTION

Motion Retargeting, widely used in character animations, involves editing the motion data of a source character to fit a target character. The process of creating 3D character animations typically includes 3D character modeling, human motion capture, and retargeting to the target character. Still, the high costs associated with motion capture limit its accessibility for individuals pursuing 3D character animations for personal interests. In contrast, video-sharing platforms abound with diverse motion data uploaded daily. However, no existing framework effectively harnesses this abundant monocular video movement data to generate 3D character animations.

This paper introduces a deep-learning framework dedicated to generating character animations from monocular videos. Our method simplifies the character animation creation process by extracting motion data from monocular videos and automatically transferring it to various character skeleton models. However, some virtual characters deviate from the typical human structure, posing particular challenges for motion retargeting. One existing approach, the Skeleton-Aware Motion Retargeting Network (SAMRN) [Aberman et al. 2020], aids in motion retargeting for characters with differing joint counts, utilizing motion capture datasets. Nevertheless, motion data directly extracted from videos through pose estimation often contains significant noise and may not align with the data structure required by the retargeting network.

In our paper, we make the following contributions:

- We devise a method to quickly generate training datasets for SAMRN from monocular videos, encompassing an extensive volume of stable motion data with minimal noise.
- We present a method to resolve the problem of artifacts in root rotation data when employing SAMRN.
- We introduce two loss functions that leverage 2D motion data estimated from videos to create character animations that closely match the original videos.

We validate the effectiveness of our proposed method through qualitative and quantitative evaluations conducted in our experiments.

2 RELATED WORK

In this section, we will explore two networks related to our method: a pose estimation network and a motion retargeting network.

2.1 Pose Estimation from Monocular Videos

Here, we will review pose estimation from monocular images or videos, specifically focusing on deep neural network approaches.

2D pose estimation can be categorized into two approaches: top-down and bottom-up. The top-down method involves detecting humans in the images and performing 2D keypoint detection for each individual [Chen et al. 2018]. In contrast, the bottom-up

method first detects all keypoints in the image and then groups them for each individual [Cao et al. 2021]. Regarding 3D pose estimation, recent years have seen the emergence of various methods, particularly in the realm of deep learning [Zheng et al. 2023]. These methods can be broadly classified into two types. The first type is an end-to-end approach that directly estimates 3D poses from images and videos [Kocabas et al. 2020; Pavlakos et al. 2018; Tripathi et al. 2023]. The second type involves estimating 2D poses from images and videos and then converting them into 3D poses. Numerous methods exist for this mapping from 2D to 3D poses [Gong et al. 2023; Li et al. 2022; Martinez et al. 2017; Zhang et al. 2022].

In our proposed framework, we present a fast and automated method for generating high-quality datasets tailored for training SAMRN. This approach leverages the 2D pose estimator OpenPose [Cao et al. 2021] and the 3D pose estimator developed by Pavllo et al. [Pavllo et al. 2019].

2.2 Motion Retargeting

In the early days of motion retargeting, numerous methods necessitated the manual design of kinematic constraints for target actions, which made the process intricate [Gleicher 1998; Lee and Shin 1999; Tak and Ko 2005]. However, in recent years, we have witnessed a surge in motion retargeting methods that harness deep learning techniques [Jang et al. 2018; Villegas et al. 2018]. Nonetheless, these methodologies have not yet tackled the challenges posed by discrepancies in skeleton structures between the source and target characters.

The Skeleton-Aware Motion Retargeting Network (SAMRN), as introduced by Aberman et al. [Aberman et al. 2020], capitalizes on its ability to map different homeomorphic skeletons to a shared primal skeleton using pooling operations. This capability simplifies the process of retargeting for characters with varying joint counts. Nevertheless, SAMRN may encounter challenges when applied to video data. Specifically, the retargeted motion generated by SAMRN can appear unnatural if the character's rotation exceeds a certain threshold, as illustrated in Figure 2. To address this issue, we propose a framework that can adapt motion data from monocular videos to suit SAMRN and introduce a loss function aimed at mitigating the rotation problem.

3 METHOD

3.1 Overview

The overall framework of our motion retargeting network, which transforms monocular videos into animations for various characters, is depicted in Figure 1. In the training phase, we initiate a two-stage pose estimation process on multiple monocular videos (Section 3.2). In the first stage, we extract actor keypoints in each frame through 2D pose estimation, perform denoising, and choose motion data based on joint position confidence values generated by OpenPose [Cao et al. 2021]. The preprocessed 2D poses are then converted into 3D motion data using VideoPose3D [Pavllo et al. 2019]. This step involves adjusting the data structure of the 3D motion data generated by VideoPose3D and aligning it with the rest pose of the target character. The resulting 3D motion data, along with the motion data from the existing character animation dataset Mixamo [Inc. 2023], are used as the source character and target character motion data for training SAMRN, respectively. To tackle the rotation issue in SAMRN, we introduce a root node rotation loss function aimed at mitigating artifacts in the root rotation data present in the retargeted motion (Section 3.3). In contrast to SAMRN, which takes 3D motion data as both input and output, our approach relies on video data as input. Consequently, we incorporate a 2D root and end-effector velocity loss function to generate motion data that closely corresponds to the original video, utilizing the preprocessed 2D poses. Additionally, we introduce a balance loss function to enhance the naturalness of the character's movements.

In the testing phase, both the motion data employed during the training phase and the motion data estimated from new videos are fed into the pretrained retargeting network as input.

3.2 Preprocessing Data Derived from Videos

To train SAMRN, we first create a training dataset, which consists of source motion data derived from multiple monocular videos through pose estimation and the motion data of the target character in the existing animation dataset Mixamo. It's important to note that the video-derived data may contain noise and unrealistic poses that could impact SAMRN's training process. Futhermore, to use pose estimation data from a video as input for SAMRN, the data structure and the character's rest pose must match the network and the target character motion data. Therefore, we utilize the pose estimator VideoPose3D, which fulfills the requirements of SAMRN's learning data.

In our framework, we employ a two-stage pose estimator composed of a 2D pose estimator and a 3D pose estimator. Due to the extensive time required for manual selection of data from videos to create a dataset of tens of thousands of frames, we perform an automated selection and denoising procedure prior to generating the 3D results from the 2D poses using detection confidence values outputted by OpenPose.

For joints with a detection confidence value of 0.3 or less, we compute the joint positions using interpolation from adjacent frames with relatively high detection confidence. The choice of 0.3 as the threshold was based on our experiments. Additionally, when encountering a sequence of more than ten frames containing a significant number of joints with low detection confidence, interpolation becomes challenging. As a result, we remove these frames and split the motion data. After producing the 3D motion data, we adjust the skeleton's size and the initial state of the source character's rest pose to match the target character before training.

3.3 Adding Loss Functions

The intrinsic loss function of SAMRN calculates the difference between the retargeting results produced by the retargeting network and the source motion corresponding to these results. Considering the inclusion of monocular video as the network input and the rotational challenges within SAMRN, we will now describe the distinct loss functions employed in our adaptation of SAMRN.

3.3.1 Root Rotation Loss. As mentioned earlier, SAMRN encounters a rotation issue where character movements appear unnatural when they undergo rotations exceeding 180 degrees, as depicted in Figure 2. We hypothesize that this issue arises from the absence of a

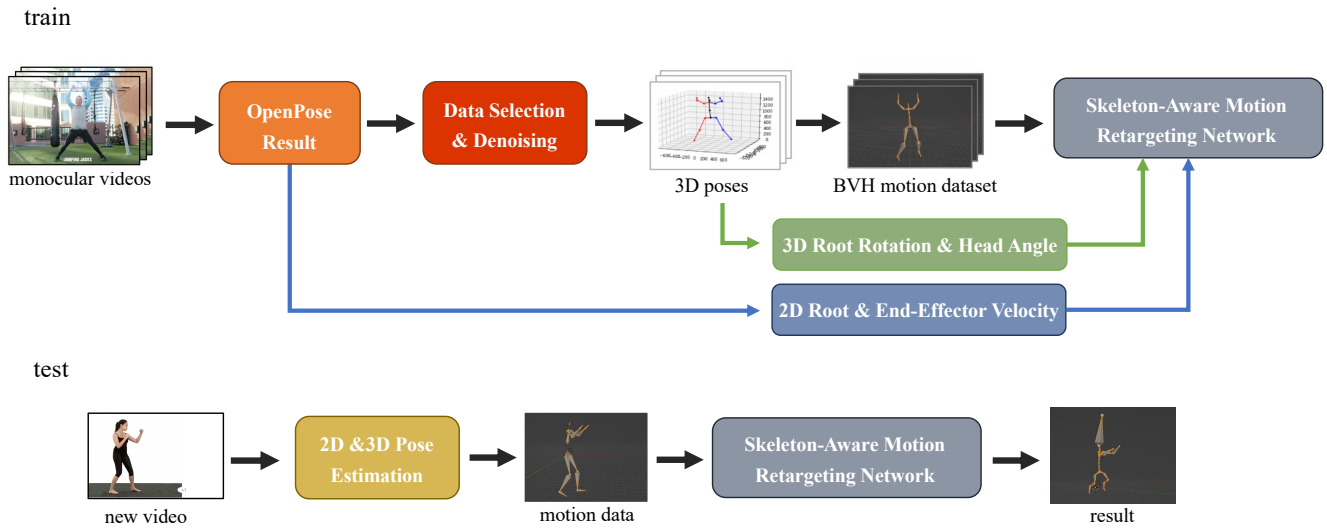

Figure 1: Overview of our method.

Figure 2: The keyframes depict a character making a 180-degree turn. The first row represents the source motion, while the second row displays the retargeted motion.

comparative analysis of the root rotation data between source and retargeted motion during the network learning process. Therefore, we introduce a loss term for the root node's rotation. Since the accuracy of root rotation data may affect the motion data of other joints in the generator of SAMRN, we employ a loss function to address the rotation issue rather than aligning the root rotation data between source and retargeted motion.

$$L_{rootrot} = \sum_t \left\| quat2eul(\hat{q}_{t,root}) - e_{t,root} \right\|, \tag{1}$$

where t denotes the frame ID, $\hat{q}_{t,root}$ signifies the rotational angle of the root node represented by a unit quaternion in the retargeting result at frame t. The term $e_{t,root}$ represents the rotation angle of the root node as delineated in Euler angles within the source motion. $quat2eul(\cdot)$ denotes the function of transforming quaternions into Euler angles.

3.3.2 2D Root & End-Effector Velocity Loss. SAMRN utilizes 3D motion data exclusively; video data, the primary input data, is not used in the retargeting network. To address the loss of motion information when generating 3D motion data from the video, it's crucial to incorporate video data into SAMRN's learning process. To achieve this, we apply two essential loss terms: a 2D root velocity loss and a 2D end-effector loss. These losses guarantee that the retargeted motion preserves the same motion features for both the root node and end-effectors, as seen in the source video.

$$L_{2Droot} = \sum_t \left\| \Pi \left[\frac{FK(\hat{q}_{t,root}) - FK(\hat{q}_{t-1,root})}{h_{3D}} \right] - \left(\frac{p_{t,2Droot} - p_{t-1,2Droot}}{h_{2D}} \right) \right\|, \tag{2}$$

$$L_{2Dee} = \sum_j^{ee} \sum_t \left\| \Pi \left[\frac{FK(\hat{q}_{t,j}) - FK(\hat{q}_{t-1,j})}{h_{3D}} \right] - \left(\frac{p_{t,2Dj} - p_{t-1,2Dj}}{h_{2D}} \right) \right\|, \tag{3}$$

where $\hat{q}_{t,j}$ symbolizes the rotation angle of joint j represented by a unit quaternion in the retargeted result at frame t, and $p_{t,2Droot}$ denotes the 2D root node position extracted from the video at frame t. The parameters h_{2D} and h_{3D} represent the character's height, which corresponds to the bone length from the head joint to the foot joint in both 2D and 3D. The function $FK(\cdot)$ signifies the forward kinematics function, while $\Pi(\cdot)$ represents the projection function used to convert from 3D to 2D.

3.3.3 Balance Loss. The position coordinates of the root node in the 3D motion estimated from videos are derived from 2D poses and lack depth information. Consequently, the character may take unbalanced poses, such as forming an improbable angle with the ground. To address this, we introduce a balance loss function to maintain the equilibrium of the character during retargeting.

$$L_{balan} = \sum_t \left\| (FK(\hat{q}_{t,head}) - FK(\hat{q}_{t,root})) - (FK(q_{t,head}) - FK(q_{t,root})) \right\|, \tag{4}$$

where $\hat{q}_{t,head}$ and $q_{t,head}$ represent the rotation angle of the head joint in the retargeted motion and the source motion at frame t, respectively. The term $FK(\hat{q}_{t,head})$ calculates the position coordinates of the head joint, while $FK(\hat{q}_{t,root})$ represents the position coordinates of the root node. Thus, the directional vector of the character's head relative to the root node in the retargeting result is given by $FK(\hat{q}_{t,head}) - FK(\hat{q}_{t,root})$.

3.3.4 Total Loss. Consequently, we provide the full loss function used for training below.

$$L = L_{origin} + \omega_{rootrot} L_{rootrot} + \omega_{2Droot} L_{2Droot} + \omega_{2Dee} L_{2Dee} + \omega_{balan} L_{balan}, \tag{5}$$

where L_{origin} denotes the loss function of the original SAMRN. During experiments with data extracted from videos, we assigned weights as $\omega_{2Droot} = 1000$, $\omega_{2Dee} = 100$, $\omega_{balan} = 300$. The root rotation loss was not used in video-based motion retargeting training due to slight unnaturalness. However, we conducted separate experiments on the rotation issue using 3D motion data with $\omega_{rootrot} = 20$ to demonstrate the effectiveness of the root rotation loss.

4 RESULTS AND DISCUSSION

4.1 Experimental Setting

In this section, we will introduce the experimental environment and data used for our approach. The proposed method was implemented in Python, utilizing the PyTorch framework for training our network. The computational resources utilized for the experiments included two NVIDIA® RTX™ 2080Ti GPUs. It is worth mentioning that the training process, which spanned 2,000 epochs, took approximately two hours.

For the experimental data, we utilized a dataset consisting of a total of 100,000 frames capturing the movements of both the source and target characters, which were derived from videos. The motion data for the source character consists of 40,000 frames obtained through pose estimation from over 20 distinct monocular videos featuring individual subjects. Notably, some of these videos are sourced from SFV [Peng et al. 2018]. The motion data for the target characters was sourced from the online platform Mixamo, totaling 60,000 frames of motion data. Throughout the learning process, a batch size of 256 was used, and the training was carried out for 2,000 epochs.

To evaluate the effectiveness of the root node rotation loss function, we trained networks exclusively utilizing this loss function. The training data consisted of 120,000 frames of motion data from Mixamo, used for both the source and target character datasets. The training process comprised 5,000 epochs.

4.2 Qualitative Evaluations

Figure 3 showcases the retargeting results achieved through the application of our proposed method on video data. To further assess the quality of the retargeting results, we encourage readers to refer to our supplementary video, which provides a qualitative evaluation of the retargeted motions generated by our method. The results demonstrate that the characters faithfully replicate the movements of the individuals in the original video. These findings underscore the effectiveness of our proposed method in naturally retargeting motion from videos onto characters. However, it is important to note that since the proposed method generates skeleton animations without considering the character's model, certain results may exhibit issues of self-contact. Additionally, it is evident that there is an issue where the feet do not make proper contact with the ground. We present the effectiveness of the proposed framework, including the retargeting results when directly applied to videos using SAMRN, as well as the impacts of implementing various loss functions in the supplementary video. In Figure 4, it is evident that with the incorporation of the root rotation loss function, the character's 180-degree rotation no longer exhibits the common rotation issue observed in SAMRN.

4.3 Quantitative Evaluations

Due to the absence of correct motion data for distinct characters corresponding to monocular videos, we employ two metrics, *relative distance metric* and *relative velocity metric*, for quantitative assessment comparing the movement characteristics between the source video and the retargeted motion. The quantitative evaluations aim to measure the similarity between the retargeted motion and the original videos.

4.3.1 Relative distance metric. We normalize the preprocessed 2D pose extracted from the video, which had undergone denoising and selecting, and the corresponding 3D pose from the retargeting results in each frame based on the length of the bones. The precision of the results is then evaluated by measuring the similarity of their end-effectors. We then compute the evaluation metric E_{dis} for relative distance as follows:

$$l_{t,j} = \Pi \left[\frac{FK(\hat{q}_{t,j}) - FK(\hat{q}_{t,root})}{KC_{root,j,3D}} \right], \tag{6}$$

$$L_{t,j} = \frac{p_{t,j} - p_{t,2Droot}}{KC_{root,j,2D}}, \tag{7}$$

$$Prop_{t,j} = \begin{cases} \frac{l_{t,j}}{L_{t,j}} & \left(\frac{L_{t,j}}{l_{t,j}} \leq 1\right) \\ \frac{L_{t,j}}{l_{t,j}} & \left(\frac{L_{t,j}}{l_{t,j}} > 1\right) \end{cases}, \tag{8}$$

$$E_{dis} = \frac{1}{Tee} \sum_{j}^{ee} \sum_{t} Prop_{t,j}, \tag{9}$$

where $l_{t,j}$ denotes the relative distance from the root node to end-effector j in the t-th frame of the 3D motion, while $L_{t,j}$ signifies the relative distance between the root node and end-effector j in the t-th frame of the 2D motion. $Prop_{t,j}$ is the ratio of the 2D to 3D relative distances, whereas $KC_{root,j,3D}$ and $KC_{root,j,2D}$ represent the lengths of the kinematic chains from the root node to end-effector j in 3D and 2D, respectively.

4.3.2 Relative velocity metric. We calculate the relative velocity of each end-effector by considering the relative distance from the root node. A smaller difference in the end-effector velocities between the 2D motion data extracted from the video and the 3D retargeted motion indicates a higher level of similarity between the two movements. We then compute the relative velocity metric E_{vel}

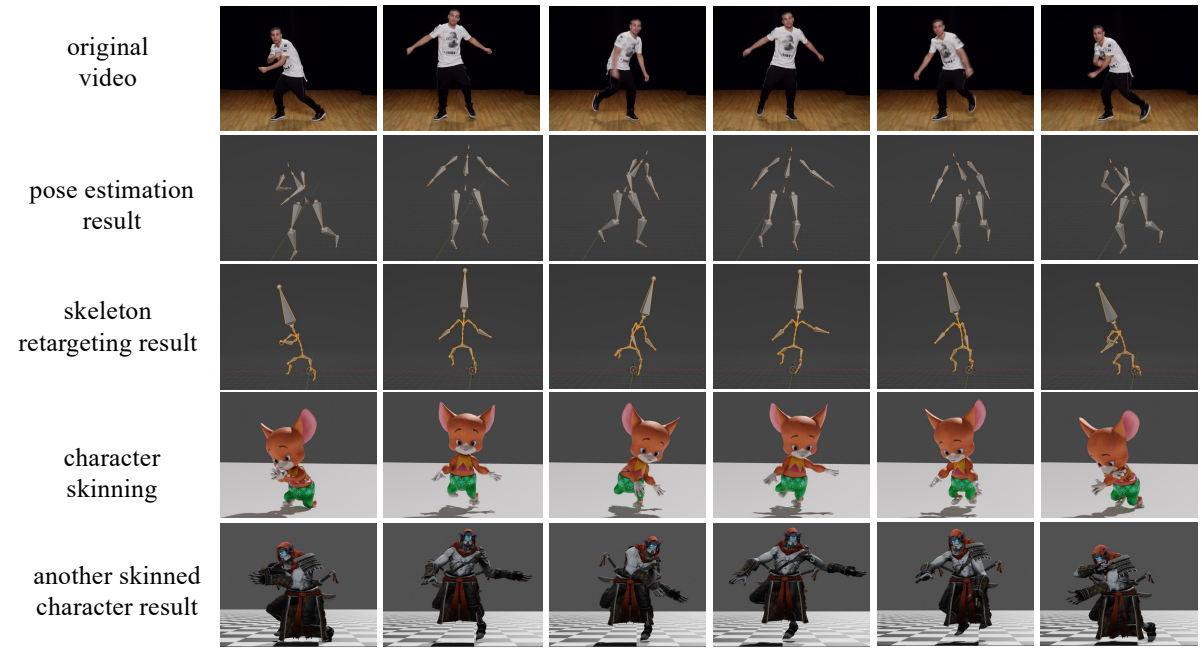

Figure 3: Motion retargeting results from a monocular video to a target character, including the motion data extracted from the video and the retargeted result visualized through skeleton animation.

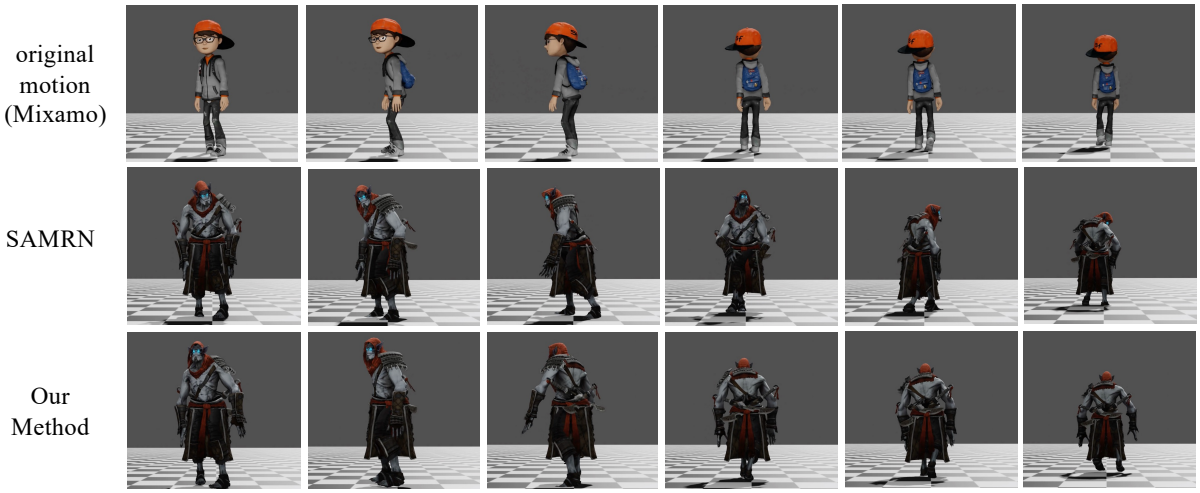

Figure 4: Motion retargeting results from 3D motion data demonstrate that the rotation issue within SAMRN can be resolved by incorporating the root node rotation angle loss function.

as follows:

$$v_{t,j} = l_{t,j} - l_{t-1,j}, \tag{10}$$

$$V_{t,j} = L_{t,j} - L_{t-1,j}, \tag{11}$$

$$E_{vel} = \frac{1}{Tee} \sum_{j}^{ee} \sum_{t} \left\| v_{t,j} - V_{t,j} \right\|, \tag{12}$$

where $v_{t,j}$ represents the relative velocity of end effector j in frame t of the 3D motion, and $V_{t,j}$ signifies the relative velocity in the 2D motion.

Table 1 displays the relative distance metric results for the representative four motions, while Table 2 showcases the relative velocity metric results. For improved clarity, the velocity results are scaled by a factor of 100. Following the retargeting of motion data extracted from 16 videos (15,000 frames) to Mixamo's character "Vampire A

Table 1: Relative distance metric.

		Jumping Jacks	Long Dance	Gymnastics	Wood Chopper
SAMRN	No Preprocessing Loss	1.44	1.52	1.77	1.52
Ours	No 2D Root & End-Effector Loss	1.38	1.30	**1.34**	**1.21**
	No Balance Loss	**1.34**	1.41	1.48	1.26
	Full Loss	1.38	**1.28**	1.50	1.26

Table 2: Relative velocity metric. (multipled by 100)

		Jumping Jacks	Long Dance	Gymnastics	Wood Chopper
SAMRN	No Preprocessing Loss	5.51	1.93	2.28	2.38
Ours	No 2D Root & End-Effector Loss	5.38	**1.67**	2.08	2.14
	No Balance Loss	**4.71**	1.41	2.07	2.18
	Full Loss	5.09	1.77	**2.06**	**2.13**

Lusth," we compute the mean joint error for the two evaluation metrics and present a subset of the corresponding results here. The full content of each motion and complete quantitative evaluation results can be found in the supplemental material.

In the quantitative evaluation, our method consistently produces character animations that exhibit a higher degree of naturalness and closely resemble real videos in comparison to applying SAMRN directly to videos. Nevertheless, it's crucial to note that when examined individually, the introduction of a loss function based on movements may inadvertently affect the performance metrics.

5 CONCLUSIONS AND FUTURE WORK

In our proposed method, we have developed a framework for retargeting motion data extracted from videos to characters with different skeletal structures. Our experimental results demonstrate that our method can generate natural character animations that effectively reconstruct the original motion of the actors. Specifically, we apply a series of preprocessing steps to the motion data obtained from multiple videos through pose estimation. To address the rotation issue, our method incorporates a loss function for the root node rotation angle in the retargeting network. Furthermore, we have demonstrated that incorporating the 2D motion information estimated from the video into the loss function of the retargeting network leads to improved retargeting results.

In future work, we aim to develop methods for retargeting videos while estimating depth information from the video itself. Another challenge that we encounter pertains to the issue of ground contact in the motion data. We have implemented a method to estimate contact labels from the video and apply inverse kinematics to the retargeted motion. However, the results indicate that this problem has not been completely resolved. Additionally, a notable limitation of our approach is that the input video must exclusively feature a single individual against a solid background.

ACKNOWLEDGMENTS

We gratefully thank the authors of SAMRN for releasing the codes, and Y. Huang, Y. Chen, K. Ni for helpful discussions. This work was partially supported by JSPS KAKENHI (Grant Num- ber JP22K12331).

REFERENCES

Kfir Aberman, Peizhuo Li, Dani Lischinski, Olga Sorkine-Hornung, Daniel Cohen-Or, and Baoquan Chen. 2020. Skeleton-Aware Networks for Deep Motion Retargeting. *ACM Trans. Graph.* 39, 4, Article 62 (aug 2020), 14 pages. https://doi.org/10.1145/3386569.3392462

Zhe Cao, Gines Hidalgo, Tomas Simon, Shih-En Wei, and Yaser Sheikh. 2021. OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. *IEEE Trans. Pattern Anal. Mach. Intell.* 43, 1 (jan 2021), 172–186. https://doi.org/10.1109/TPAMI.2019.2929257

Yilun Chen, Zhicheng Wang, Yuxiang Peng, Zhiqiang Zhang, Gang Yu, and Jian Sun. 2018. Cascaded Pyramid Network for Multi-Person Pose Estimation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE Computer Society, USA, 7103–7112. https://doi.org/10.1109/CVPR.2018.00742

Michael Gleicher. 1998. Retargetting Motion to New Characters. In *Proc. 25th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '98)*. ACM, New York, NY, USA, 33–42. https://doi.org/10.1145/280814.280820

Jia Gong, Lin Geng Foo, Zhipeng Fan, Qiuhong Ke, Hossein Rahmani, and Jun Liu. 2023. DiffPose: Toward More Reliable 3D Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 13041–13051.

Adobe Inc. 2023. Mixamo dataset. https://www.mixamo.com/ accessed: 13th July, 2023.

Hanyoung Jang, Byungjun Kwon, Moonwon Yu, Seong Uk Kim, and Jongmin Kim. 2018. A Variational U-Net for Motion Retargeting. In *SIGGRAPH Asia 2018 Posters*. ACM, New York, NY, USA, Article 1, 2 pages. https://doi.org/10.1145/3283289.3283316

Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. 2020. VIBE: Video Inference for Human Body Pose and Shape Estimation. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, USA, 5252–5262. https://doi.org/10.1109/CVPR42600.2020.00530

Jehee Lee and Sung Yong Shin. 1999. A Hierarchical Approach to Interactive Motion Editing for Human-like Figures. In *Proc. 26th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '99)*. ACM Press/Addison-Wesley Publishing Co., USA, 39–48. https://doi.org/10.1145/311535.311539

Wenhao Li, Hong Liu, Hao Tang, Pichao Wang, and Luc Van Gool. 2022. MHFormer: Multi-Hypothesis Transformer for 3D Human Pose Estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 13147–13156.

Julieta Martinez, Rayat Hossain, Javier Romero, and James J. Little. 2017. A Simple Yet Effective Baseline for 3D Human Pose Estimation. In *Proc. IEEE International Conference on Computer Vision (ICCV)*. IEEE, USA, 2659–2668. https://doi.org/10.1109/ICCV.2017.288

Georgios Pavlakos, Xiaowei Zhou, and Kostas Daniilidis. 2018. Ordinal depth supervision for 3d human pose estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 7307–7316.

Dario Pavllo, Christoph Feichtenhofer, David Grangier, and Michael Auli. 2019. 3D Human Pose Estimation in Video With Temporal Convolutions and Semi-Supervised Training. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, USA, 7753–7762. https://doi.org/10.1109/CVPR.2019.00794

Xue Bin Peng, Angjoo Kanazawa, Jitendra Malik, Pieter Abbeel, and Sergey Levine. 2018. SFV: Reinforcement Learning of Physical Skills from Videos. *ACM Trans. Graph.* 37, 6, Article 178 (Nov. 2018), 14 pages.

Seyoon Tak and Hyeong-Seok Ko. 2005. A Physically-Based Motion Retargeting Filter. *ACM Trans. Graph.* 24, 1 (jan 2005), 98–117. https://doi.org/10.1145/1037957.1037963

Shashank Tripathi, Lea Müller, Chun-Hao P. Huang, Omid Taheri, Michael J. Black, and Dimitrios Tzionas. 2023. 3D Human Pose Estimation via Intuitive Physics. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 4713–4725.

Ruben Villegas, Jimei Yang, Duygu Ceylan, and Honglak Lee. 2018. Neural Kinematic Networks for Unsupervised Motion Retargetting. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. CVF / IEEE, USA, 8639–8648. https://doi.org/10.1109/CVPR.2018.00901

Jinlu Zhang, Zhigang Tu, Jianyu Yang, Yujin Chen, and Junsong Yuan. 2022. MixSTE: Seq2seq Mixed Spatio-Temporal Encoder for 3D Human Pose Estimation in Video. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 13232–13242.

Ce Zheng, Wenhan Wu, Chen Chen, Taojiannan Yang, Sijie Zhu, Ju Shen, Nasser Kehtarnavaz, and Mubarak Shah. 2023. Deep learning-based human pose estimation: A survey. *Comput. Surveys* 56, 1 (2023), 1–37.